TCP FIN Flood Attack Pattern Recognition on Internet of Things with Rule Based Signature Analysis

https://doi.org/10.3991/ijoe.v15i07.9848

Deris Stiawan (✉), Dimas Wahyudi, Ahmad Heryanto, Samsuryadi
Universitas Sriwijaya, Palembang, Indonesia
deris@unsri.ac.id

Mohd. Yazid Idris, Farkhana Muchtar
Universiti Teknologi, Kaula Lumpur, Malaysia

Mohammed Abdullah Alzahrani
Ministry of Communications and IT, Riyad, Saudi Arabia

Rahmat Budiarto
Albaha University, Albaha, Saudi Arabia

Abstract—Focus of this research is Transmission Control Protocol (TCP) FIN flood attack pattern recognition in Internet of Things network using rule based signature analysis method. Dataset is created using three traffic scenarios: normal, attack and normal-attack. The process of identification and recognition of TCP FIN flood attack pattern is done by observing and analyzing packet's attributes from raw data (pcap format) through a feature extraction and feature selection processes. Further experiments were conducted using Snort as intrusion detection system (IDS). The evaluation results of the rate of confusion matrix detection against the Snort as IDS show the average percentage of the precision level.

Keywords—Internet of Things (IoT), TCP FIN flood attacks, Denial of Service, rule-based, signature analysis, confusion matrix.

1 Introduction

Internet of Things (IoT) is a network which integrates various identification, sensing and communication technology devices such as Radio Frequency Identification (RFID), tags, sensors, actuators, cameras, mobile phones, and various wire/wireless devices via a unique addressing schema based on standard communication protocol [1]. Each object in the IoT network is capable to interact, work together, processing and delivering information autonomously to produce services, such as statistical information, monitoring and control systems [2]. IOT is classified into three layers, which are Application Layer, Network Layer and Perception Layer [3], [4]. Main challenge in implementation of IoT is security issue,

such as privacy, authorization, verification, system configuration, access control, storage and information management [5]. Meanwhile, Denial of Service attacks (DoS) is one of the security threats on IoT network. DoS is defined as one of attacking method by attacker to spend resources, such as bandwidth and increasing energy consumption which results in energy source on the device will be quickly exhausted [6],[7]. Research work in [8], explained DoS attacks can be grouped into two main categories, which are (i) DoS Flooding Attack is defined as an attack with technique of sending many packets to the target with aim to keep the function of the CPU, memory and network resource is not optimal. (ii) Logic attack defined as attack by taking advantage of existing weaknesses to cause system malfunction.

TCP connection model uses two control flags, the SYN and FIN. Both normally are not set in the same TCP segment header (See Figure 1). The SYN flag synchronizes sequence numbers to initiate a TCP connection. The FIN flag indicates the end of data transmission to finish a TCP connection. Their purposes are mutually exclusive. A TCP header with the SYN and FIN flags set is anomalous TCP behavior, causing various responses from the recipient, depending on the operating system (OS). A FIN scan is a type of scan whose usual aim is to perform network reconnaissance. What is attractive about A FIN scan from the attacker's point of view is that the attacker sends a special signal (a TCP packet with only the FIN flag set) that tends to get past many firewalls.

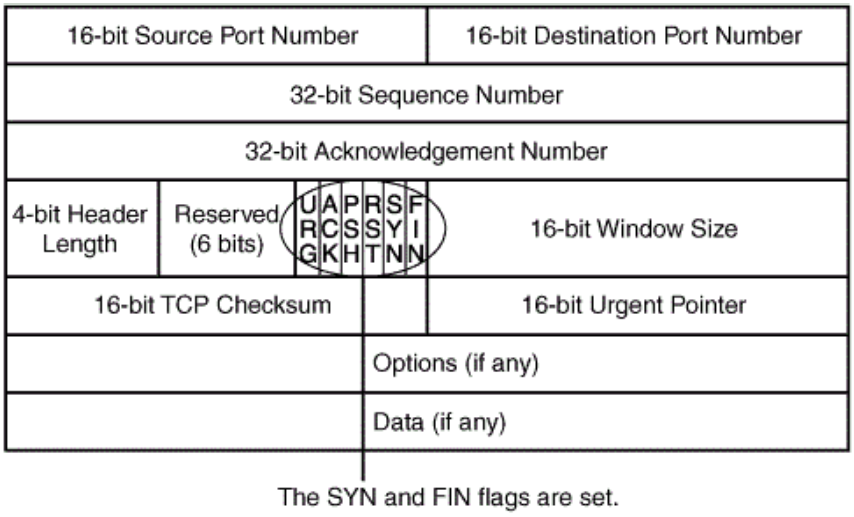

Fig. 1. TCP Header with SYN and FIN Flags set.

To address this casualty of the DoS attack, this paper attempts to come up with a strategy to detect the TCP FIN-based DoS flooding attacks in IoT. Therein answered how to identify those patterns of the attack using a rule-based signature analysis on WiFi communications. The main contribution of this paper is a strategy to analyze the TCP FIN DoS attack by characterizing the attack patterns thru thresholding deployment of IoT dataset and varying the attacks on the IoT network traffic, in contrast to

the previous works that focus on TCP SYN message to analyze TCP flood DoS attacks. Thus, this work is the first work that characterizes and analyzes the patterns of TCP FIN DoS attacks and uses the characteristics for generating the rules for detection.

The paper is arranged into five sections as follows. Section 2 discusses related works on overlay gaps of IoT threats and the assignment problem along with their current uses. Section 3 presents the experimental scenario. The results and discussion are described in Section 4. The paper ends with a conclusion and future works delivered in Section 5.

2 Related Work

Research works in [5] and [9], discusses about security issue on the nodes (sensors or controllers) that use Radio Frequency (RF) communication protocol such as WiFi, RFID, IEEE 802.15.4/ZigBee and bluetooth which generally apply broadcast mechanism to communicate with each other. This Mechanism is difficult to protect from the attack. The node on IoT is susceptible to various types of threats and attacks that include capturing, eavesdropping and tampering. Limited resources on a node are utilized by performing DoS attacks, such as DOS flooding which causing the node performs at their maximum ability that consumes its energy as well as bandwidth.

Authors in [10] discusse three types of DoS attacks on IoT nodes, which are ICMP flood, SYN flood and TCP flood. The authors compare the three types of the attacks by considering parameters: CPU utility, memory utility, delay time and packet loss rate.

There are many research works on TCP SYN flood attacks such as [11], [12], [13], however very few research works on TCP FIN flood attacks. Yoon et al. [14] discuss defense against general TCP Flooding Attack including the TCP FIN attacks. The authors describe the TCP FIN attacks in detail using state transition diagram. After a TCP session connection is created, a FIN packet or a Reset (RST) packet is instantly transmitted while data packet is not transmitted, so that the session is terminated, thereby adding to the load of the server. The connection flooding attack is able to be detected when the number of sessions, in which the FIN packet or the RST packet is received in the session state “waiting for FIN packet” or “waiting for RST packet”, is equal to or larger than a threshold.

Then, proposal works by [6] and [15] discuss the mechanism implementation of the attacks detection in an Intrusion Detection System (IDS) on IoT network by using rule based. In these works the IDS is distributed among a group of nodes in the network to avoid problems related to the limited resource on IoT devices.

3 Experiment Scenario

This research uses rule based signature analysis method to identify and recognize the type of DoS attack patterns in the form of TCP FIN flood attacks on IoT network. Several stages involve in this experiments:

- The design of a testbed system for the IoT network
- Running experiments on the testbed network with a normal scenario, attack scenario, and normal attack scenario for the purpose of a dataset creation
- Feature Extraction
- Identification of TCP FIN flood attack patterns

3.1 Testbed network design stage

The testbed network is developed by the following steps: designing the topology, hardware requirement identification, software requirement identification, installation and system configuration, and then experimenting some scenarios for creating dataset.

The testbed network consists of multiple hardware including DHT22 sensor, MQ2 sensor, soil moisture sensor, water level sensor, and WeMos D1 microcontroller equipped with ESP8266 WiFi module. In addition, the testbed utilizes supporting software such as MySQL database, DoS tools Hping3, Apache Web Server and Snort as IDS. Hping3 injects the TCP FIN flood attacks to the testbed network. Figure 2 illustrates the topology of the testbed network.

As shown in Figure 2, the testbed network topology consists of four sensors nodes, one server, and two laptops as sniffing and attacker.

The type of topology is star topology where each sensor node and the server are connected in one network via wireless router with Dynamic Host Configuration Protocol (DHCP) for IP address configuration.

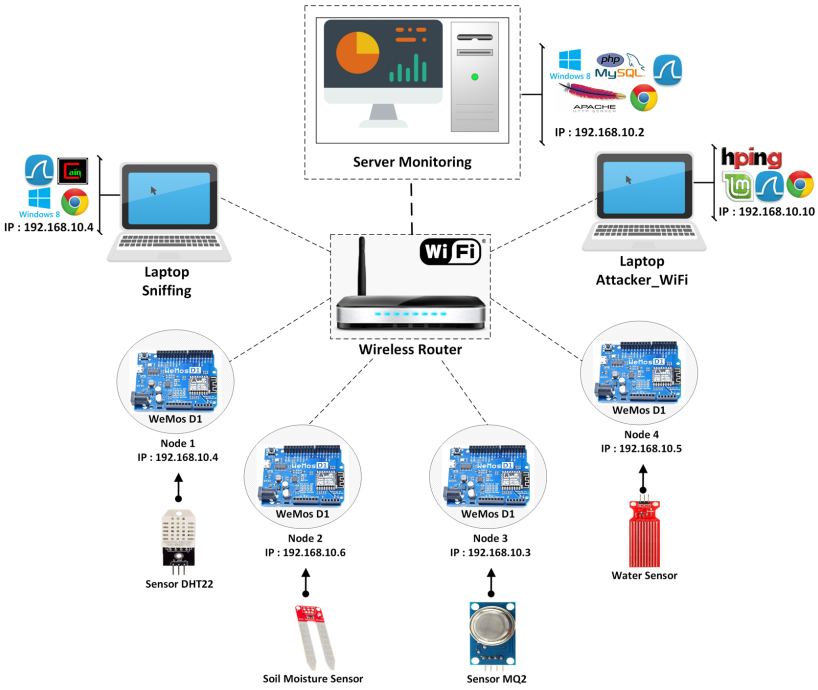

Fig. 2. Testbed network topology

3.2 Dataset creation stage

Dataset creation in this research was done by running three scenarios:

- Normal traffic
- TCP FIN flood attack traffic
- Normal data-TCP FIN flood attack traffic

Each scenario of the dataset creation was conducted for five minutes at sensor node 1 to sensor node 4 and the server. Sniffer modules capture the traffic packets and save them as a raw data in pcap format. Then, the next stage; feature extraction is conducted with the aim to get detail information from the generated dataset. This stage is one part of identification process of TCP FIN flood attack pattern based on observation and analysis toward package attributes from raw data (pcap format).

3.3 Feature extraction stage

Figure 3 shows the flowchart of feature extraction process. The attributes used in this process include frame.number, frame.time, frame.len, ip.src, ip.dst, tcp.srcport, tcp.dstport, tcp.ack, tcp.hdr_len, tcp.window_size_value, ip.protocol, ip.flags, ip.len, ip.TTL. A converter module changes the pcap format file into a CSV (Comma Separated Value) format.

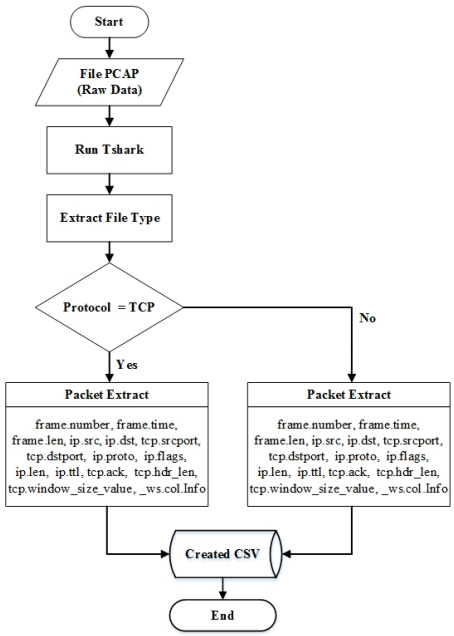

Fig. 3. Flowchart for feature extraction

3.4 TCP FIN Flood attack pattern detection stage

Attack patterns identification is conducted to recognize patterns which have already known/recognized (as a signature) through the following steps: analysis of the raw data (pcap) normal packets compared to attack packets, testing dataset with Snort as IDS, and the analysis of the correlation between Snort alert logs from the raw data (pcap format) and feature extraction results from CSV type file. The TCP FIN attack detection engine will be using the recognized patterns as a basis for its rule-based.

3.5 Performance evaluation

The IDS is expected to maximize the detection accuracy of the existence of attacks (true positive) and at the same time to reduce false detection where a normal network traffic is indicated as an attack (false positive). Sometimes it may happen the IDS fails to give alert of attack which occurred (false negative), or if an attack occurs and the system alarm detection does not appear (true negative).

There are seven performance indicators of IDS. They measure the level of accuracy, detection rate, false alarm rate, and the rate of precission as represented in (1) to (7). This work uses these indicators.

$$\text{TPR} = \frac{TP}{TP + FN} \tag{1}$$

$$\text{FPR} = \frac{FP}{TN + FP} \tag{2}$$

$$\text{TNR} = \frac{TN}{TN + FP} \tag{3}$$

$$\text{FNR} = \frac{FN}{TP + FN} \tag{4}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Non} - \text{Precision} = \frac{TN}{TN + FN} \tag{6}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

4 Experiment Results and Discussion

The result of running the testbed network topology in Figure 2 creates six datasets with two different types of data packets: normal data packets and TCP FIN flood attack data packets as shown in Table 1.

Table 1. Dataset creation results

No.	Dataset (Label)	Size	Note
1.	normal_server.pcap testbed 1	1,8 MB	Server
2.	attack_server.pcap testbed 2	220,2 MB	Server
3.	normalxattack_server.pcap testbed 3	261,5 MB	Server
4.	normal_node_wifi.pcap testbed 4	1,1 MB	Node 1-4
5.	attack_node_wifi.pcap testbed 5	168,4 MB	Node 1-4
6.	normalxattack_node_wifi.pcap testbed 6	170,3 MB	Node 1-4

Experimental results are categorized based on the attack objects either server or sensor nodes. The results shown in Table 1 show the significance size changes on each experiment category. In the category of attack with the server as the target object, the size changes happened in experiment #2 and experiment #3. Whereas for the attack with the sensor nodes as the targeted objects, the size changes happened, in

experiment #5 and experiment #6. These facts show that TCP FIN flood attack uses a lot of resources on the targeted object.

4.1 Dataset analysis

The calculation of the number of packets on the dataset is done based on category of the used protocols. The experimental results show the number of TCP data packets is significantly larger compared to the other data packets. Table 2 shows the calculation results of the number of data packets. The highest percentage of the packet number is for TCP with 98.32%, followed by Address Resolution Protocol (ARP) with 1.07%, Internet Control & Management Protocol (ICMP) with 0.72%, User Datagram Protocol (UDP) with 0.52%, and unknown protocols with 0.02%.

Table 2. Dataset on the server

Dataset	Traffic					Total
	UDP	***TCP***	***ICMP***	***ARP***	***Unknown***	
Running 1	96	11.728	500	457	11	12.792
Running 2	109	3.134.653	0	626	5	3.135.393
Running 3	107	3.708.928	0	639	7	3.709.681

Table 3 shows the highest number of packets for sensor nodes 1 to 4 as follows. TCP with 98.39%, followed by ARP with 1.07%, UDP with 0.52%, and unknown protocols with 0.02%.

Table 3. Dataset on sensor node 1 – 4.

Dataset	Traffic					Total
	UDP	***TCP***	***ICMP***	***ARP***	***Unknown***	
Running 4	120	7.441	0	241	4	7.806
Running 5	222	2.397.624	0	1.564	10	2.399.420
Running 6	272	2.424.685	0	1.640	2	2.426.599

The analysis on the elaboration of protocol category of the captured data packets resulting in domination of TCP packets in each of the experiment on server as the target object as well as sensor nodes as the target objects and reached up to 98%. The huge number of TCP packets is an initial observation that indicates there are already packets from TCP FIN flood attacks on the testbed network of experiment scenario #2, #3, #5 and #6.

4.2 Feature extraction analysis

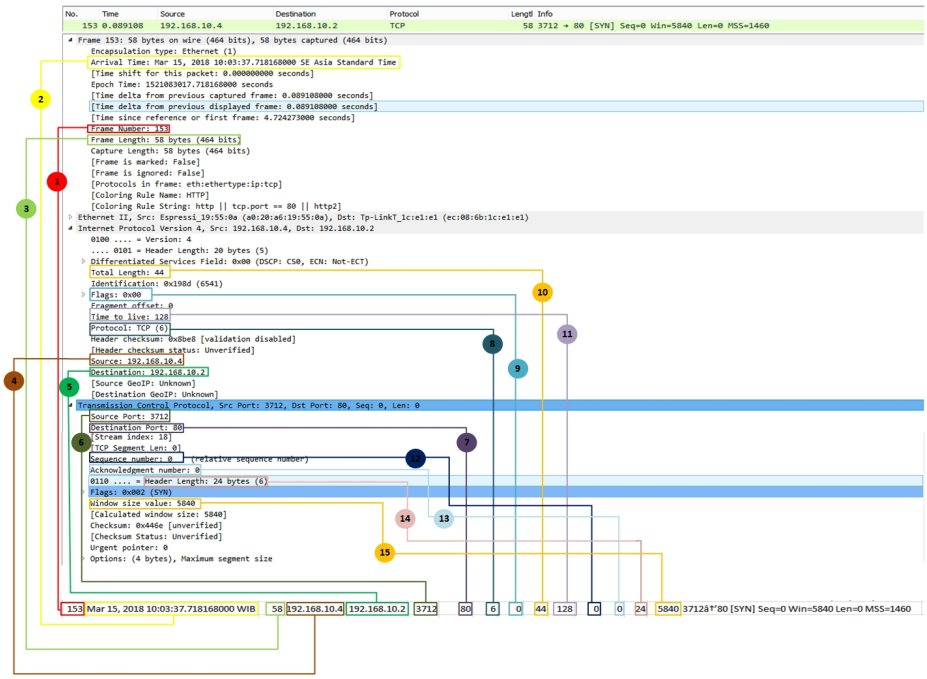

Fig. 4. Data correlation between feature extraction and raw data (pcap).

Having done the initial observation, the next stage is to perform data correlation analysis by comparing raw data (pcap) and the results of feature extraction by considering attributes resulted by the flowchart in Figure 3. The feature extraction results are at the bottom of Figure 4 and the information from pcap file that displayed on Snort are on the top of the figure. Findings of this analysis is the information in feature extraction process and the information in pcap file are consistent. For example, the time stamp information of the packet from feature extraction process is the same with the information in the pcap file (indicated by yellow color/ point 2). The standard rules in the Snort IDS are not accurate enough in detecting the TCP FIN attacks. Thus, the rules in the Snort IDS are customized by incorporating the rules produced by detection engine in Section 3.4.

4.3 Attack pattern analysis

Now, the running dataset shown in Table 1 is used to conduct experiment with Snort as Intrusion Detection System (IDS). Snort generates alert log, subsequently identified as attack pattern and correlation analysis was performed to validate the generated alerts. Table 4 shows the results. In Snort, a priority tag assigns a severity level to rules. A *classtype* rule assigns a default priority (defined by the configuration classification option) that may be overridden with a priority rule. Thus, in this work

alerts with priority=2 are more severe than alerts with priority=3. Figure 5 depicts the priority as severity levels.

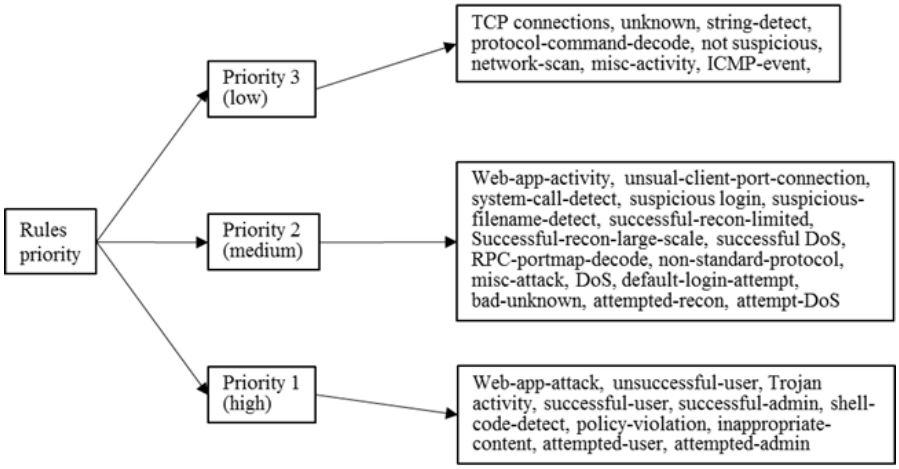

Fig. 5. Priority as severity level

Table 4 shows that the modified Snort IDS is able to detect the TCP FIN flood attacks and displays the Scan FIN alerts accurately in running experiment #2, #3, #5, and #6.

Table 4. Dataset of sensor node 1 – 4

Dataset	Alert	Priority	Total
Running 1	SCAN UPnP service discover attempt	3	250
	ICMP Echo Reply	3	250
	ICMP PING	3	250
	ICMP PING *NIX	3	250
Running 2	BAD-TRAFFIC tcp port 0 traffic	3	9
	SCAN FIN	2	540,399
Running 3	BAD-TRAFFIC tcp port 0 traffic	3	4
	SCAN FIN	2	251,561
Running 4	-	-	-
Running 5	BAD-TRAFFIC tcp port 0 traffic	3	7
	SCAN FIN	2	517,041
Running 6	SNMP AgentX/tcp request	2	1
	SNMP trap tcp	2	1
	BAD-TRAFFIC tcp port 0 traffic	3	9
	SCAN UPnP service discover attempt	3	13
	SCAN FIN	2	555,779

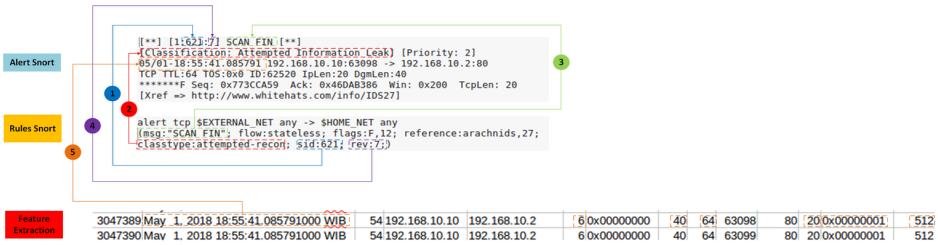

Fig. 6. Matching alert and rules of Snort against feature extraction results.

Figure 6 exhibits the data correlation analysis to validate the alert log generated by Snort. Similar with explanation of Figure 4, the alert information from feature extraction process are displayed at the bottom of the figure, while the upper part of the figure are alert information from the Snort IDS. Hence, the TCP FIN flood attack patterns were defined as rules as shown in Table 5.

The pattern of attacks in Table 5 are defined as rules which required as knowledge based of attacks, patterns and filtered data for the modified Snort intrusion detection engine.

Table 5. TCP FIN flood attack pattern.

Dataset	IP TTL	IP hdr length	IPlength	TCP flag	Win size	TCP hdr length
Testbed 1 Normal	64	20	40	" F " 0x001 (FIN)	0x200 (16) 512 (10)	20
Testbed 2 Attack	64	20	40	" F " 0x001 (FIN)	0x200 (16) 512 (10)	20
Testbed 4 Normal	64	20	40	" F " 0x001 (FIN)	0x200 (16) 512 (10)	20
Testbed 5 Attack	64	20	40	" F " 0x001 (FIN)	0x200 (16) 512 (10)	20

4.4 Results analysis

Having done running the experiments on Snort-based IDS, an assessment is conducted on the total alerts (TP, FP, TN, and FN) by the use of confusion matrix. The assessment results are in the form of binary classification, detection rate and the level of detection accuracy.

Based on the information in Table 6, binary classification in running 2 shows the number of successfully detected attacks (TP) is 540,408 (0.173%), The number of normal packages classified as attacks (False Potive) = 9 packets (0.1 %). Alarms/alerts did not appear when attacks happened is 2,578,383 packets (0. 8253%). Alerts appear when attack did not happen is 5,436 packets (0.0017%).

Table 6. Confusion matrix calculation

Binary Classification	**Snort Alert**			
	Server		***Node WiFi***	
	Running 2	***Running 3***	***Running 5***	***Running 6***
TP	540,408	251,561	517,041	555,779
FP	9	4	7	24
FN	2,578,383	3,439,650	1,564,161	1,576,338
TN	5,436	5,515	158,223	146,303
Detection Rate	**Snort Alert**			
	Server		***WiFi Sensor Nodes***	
	Running 2	***Running 3***	***Running 5***	***Running 6***
TPR (%)	17.32748363	6.815134654	24.84338378	26.06700289
FPR (%)	0.165289256	0.072476898	0.00442394	0.016401621
TNR (%)	99.83471074	99.9275231	99.99557606	99.98359838
FNR (%)	82.67251637	93.18486535	75.15661622	73.93299711
Precision (%)	99.99833462	99.99840995	99.99864616	99.99568192
Non-Precision (%)	0.210386254	0.160079416	9.186279018	8.492947747
Accuracy (%)	17.47127938	6.954145961	30.15336032	30.81409945

Calculations of confusion matrix on the running Snort-based IDS are shown in Table 6. The calculation involves the total alert resulted from the running dataset as follows.

- On server attack dataset (running 2); 540,408 out of 3,124,236 packets are indicated as alerts or 0.1730%,
- Normal - attack server dataset (running 3) 251,561 out of 3,696,730 packets are indicated as alerts or 0.0680%,
- Node Wi-Fi attack dataset (running 5); 517,041 out of 2,239,432 packets are indicated as alerts or 0.2309%,
- Normal - Wi-Fi sensor nodes attacks dataset-running running 6); total alerts generated is 555,779 out of 2,278,444 packets or 0.2439%.

Furthermore, from the information in Table 6, binary classification in running 2 shows the number of successfully detected attacks (TP) is 540,408 (0.173%), The number of normal packages classified as attacks (FP) = 9 packets (0.0000%). Alarms/alerts did not appear when attacks happened (FN) =2,578,383 packets (0.8235%) and alerts appear when attacks did not happen (TN) =5,436 packets (0.0017%).

In running 3, on binary classification; TP = 251,561 (0.0680%), FP = 4 (0.000%), FN = 3,439,650 (0.9305%) and TN = 5,515 (0.001%).

In running 5, on binary classification; TP = 517,041 (0.2309%), FP = 24 (0.000%), FN = 1,564,161 (0.6985%) and TN = 158,223 (0.070%).

In running 6, on binary classification; TP = 555,779 (0.2439%), FP = 7 (0.000%), FN = 1,576,338 (0.6918%) and TN = 146,303 (0.064%).

Hence, TP average = 17.8958%, FP average = 0.0004%, FN average = 78.6513% and TN average = 3.4524%,

Figure 7 shows the binary classification comparison chart with the False Negative (FN) parameter has the highest average value of percentage of 78.6513%. From the four measurements, the performance of the Snort-based detection with default rules relies mainly on two aspects: True Positive and False Positive numbers.

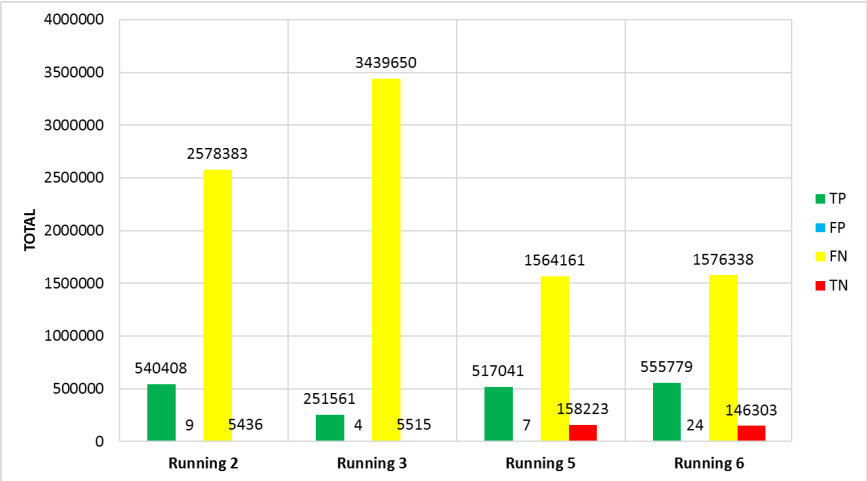

Fig. 7. Binary classification comparison graph

The following is an example on the steps of the confusion matrix calculation to measure accuracy level of the IDS against the TCP FIN flood attack using experimental data #2 of Table 6.

From (1), $TPR = \frac{TP}{TP + FN}$

Thus, $TPR = \frac{540408}{(540408 + 25783830)} = 17.32748363\ \%$

From (2), $FPR = \frac{FP}{TN + FP}$

Thus, $FPR = \frac{9}{(5436 + 9)} = 0.165289256\ \%$

From (3), $TNR = \frac{TN}{TN + FP}$

Thus, $TNR = \frac{5436}{(5436 + 9)} = 99.83471074\ \%$

From (4), $FNR = \frac{FN}{FN + TP}$

Thus, $FNR = \frac{2578383}{(2578383 + 540408)} = 82.67251637\ \%$

From (5) $Precision = \frac{TP}{TP + FP}$

Thus, $Precesion = \frac{540408}{(540408 + 9)} = 99.99833462\ \%$

$$\text{From (6), NonPrecision} = \frac{TN}{TN + FN}$$

$$\text{Thus, NonPrecesion} = \frac{5436}{(5436 + 2578383)} = 0.210386254\ \%$$

$$\text{From (7), Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Thus, Accuracy} = \frac{(540408 + 5436)}{(540408 + 5436 + 9 + 2578383)} = 17.471279\ 38\ \%$$

Therefore, from the data in Table 6, for the four running experiments we obtain the following.

- The average percentage of the True Positive Rate (TPR) is 18.7632%,
- The average of False Positive Rate (FPR) is 0.0646%,
- The average of True Negative Rate (TNR) is 99.9353%,
- The average of False Negative Rate (FNR) is 81.2367%,
- The average of precision level is 99.9977%,
- The average of non-precision level is 4.5124%, and
- The average of accuracy level is 21.3482%.

The comparison on detection rate is visualized in Figure 8 (the data is chunked with only 3 decimal points).

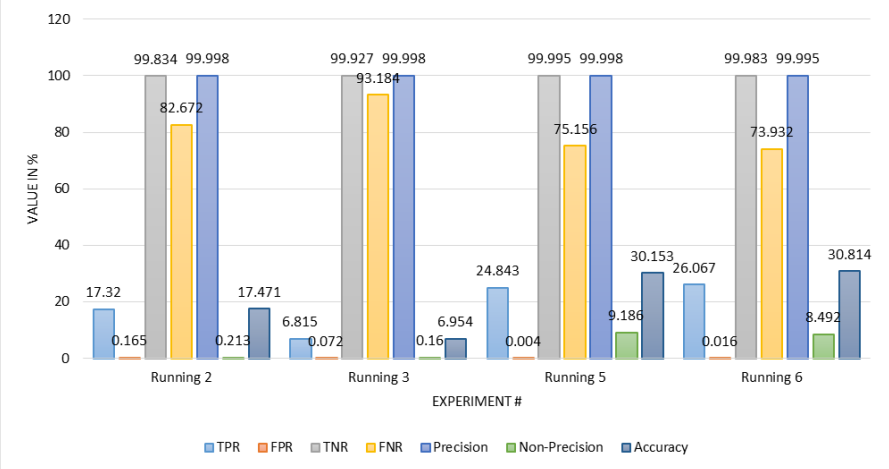

Fig. 8. Graph of detection rate comparison.

Overall, the experiment results show that the TCP flood attack detection using TCP FIN feature relatively provides better accuracy compare to the detection that use TCP SYN, because this work has successfully characterized the TCP FIN attacks then uses the characteristics for developing better rules.

5 Summary

TCP FIN flood attack pattern has attack pattern in the form of {ip.ttl: 64, ip. hdr_len: 20, ip.len:40, tcp.flags:F, window:512, tcp.hdr_len:20, wpan.src_pan: 0xffff, wpan.dst16:0x00, wpan.c_md: 0x01, data.len:1}. This pattern can be used to constructing rules in intrusion detection engine to improve accuracy of the detection.

Evaluation results of confusion matrix of the detection rate against the Snort IDS running results showed the average percentage of True Positive Rate (TPR) is 18.7632%, the False Positive Rate (FPR) is 0.0646%, True Negative Rate (TNR) is 99.9353%, False Negative Rate (FNR) is 81.2367%, the level of precision is 99.9977%, non-precision level is 4.5124% and accuracy level is 21.3482%. The results showed that the TCP DoS attack detection using TCP FIN message provides better accuracy compared to the detection using TCP SYN message. As for further research, the authors consider to make the running dataset to have more varied scenarios to generate variety attack patterns with the aim to seek more complicated attack patterns.

6 References

[1] Li S, Xu L.D, and Zhao S (2015). The Internet of Things: a survey. Information System Frontiers, 17(2):243–259. https://doi.org/10.1007/s10796-014-9492-7

[2] Atzori L, Iera A, and Morabito G (2010). The Internet of Things : a survey. Computer Networks, 54(15):2787–2805. https://doi.org/10.1016/j.comnet.2010.05.010

[3] Gubbi J, Buyya R, Marusic S, and Palaniswami M (2013). Internet of Things (IoT): a vision, architectural elements, and future directions. Future Generaation Computer Systems. 29(7): 1645–1660. https://doi.org/10.1016/j.future.2013.01.010

[4] Perera C, Zaslavsky A, Christen P, and Georgakopoulos D (2014). Context aware computing for The Internet of Things. IEEE Communication Survey and Tutorials. 16(1):414–454. https://doi.org/10.1109/SURV.2013.042313.00197

[5] Alaba F.A, Othman M, Hashem I. A. T, and Alotaibi F (2017). Internet of Things security: a survey. Journal of Network and Computer Application, 88: 10–28. https://doi.org/10.1016/j.jnca.2017.04.002

[6] Zarpelão B.B, Miani R.S, Kawakani C.T, and de Alvarenga S.C (2017). A survey of intrusion detection in Internet of Things. Journal of Network and Computer Application, 84: 25–37. https://doi.org/10.1016/j.jnca.2017.02.009

[7] Wood A.D and Stankovic J. A (2012). Denial of service in sensor networks. Computer (Long. Beach. Calif). 35(10):54–62.

[8] Wang D, He L, Xue Y, and Dong Y. Exploiting artificial immune systems to detect unknown DoS attacks in real-time, IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS 2012), October. 31- November 1 2012, vol. 2, pp. 646–650. Ghildiyal S, Mishra A. K, Gupta A, and Garg N (2014). Analysis of denial of service (DoS) attacks in wireless sensor networks. International Journal of Research in Engineering and Technology, 03(22):140–143.https://doi.org/10.15623/ijret.2014.0322030

[9] Liang L, Zheng K, Sheng Q, Wang W, Fu R, and Huang X (2017). A denial of service attack method for IoT system in photovoltaic energy system. Lecture Notes on Computer

Science (including Subser. Lecture Notes on Artificial Intelligence, Lecture Notes on Bioinformatics), 10394:613–622. https://doi.org/10.1007/978-3-319-64701-2_48

[10] Haris S.H.C, Badlishah A.R, Ghani M.A.H.A, Waleed G.M. TCP SYN flood attack analysis based on payload. IEEE Student Conference on Research and Development (SCOReD 2010), Dec.13 - 14 2010, Putrajaya, Malaysia, pp. 149-153. https://doi.org/10.1007/978-3-319-64701-2_48

[11] Bogdanoski M, Shuminoski T, Risteski A (2013). Analysis of the SYN flood DoS attack. International Journal of Computer Network and Information Security. 5:1-11. https://doi.org/10.5815/ijcnis.2013.08.01

[12] Bellaïche M and Grégoire J-C (2011). SYN flooding attack detection by TCP handshake anomalies. Security and Communication Networks, 5:709–724. https://doi.org/10.1002/sec.365

[13] Yoon S., Oh J., Kim I. and Jang J. Defense against TCP Flooding Attack. International Conference on Security and Cryptography (SECRYPT-2012), July 24-27 2012, Rome, Italy, pp.416-420. https://doi.org/10.1002/sec.365

[14] Chandulal J.A, Rao D. K. N, and Akbar S. (2010). Intrusion detection system methodologies based on data analysis. Foundation of Computer Science, 5(2): 10–20.

7 Authors

Dr. Deris Stiawan is the lab head of "Deris Stiawan Lab" and an associate professor & faculty of Computers. His main interests are Information and Communication Technology, IT Security, Intrusion Prevention, CCNA, C|EH & C|HFI

Dimas Wahyudi is a Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia.

Ahmad Heryanto is a Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia.

Samsuryadi is a Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia.

Mohd. Yazid Idris is from School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia.

Farkhana Muchtar is from School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia.

Mohammed Abdullah Alzahrani is from Technical Infrastructure Dept., Ministry of Communications & IT, Saudi Arabia.

Rahmat Budiarto is from Technical Infrastructure Dept., Ministry of Communications & IT, Saudi Arabia.

Article submitted 2018-11-12. Resubmitted 2019-03-18. Final acceptance 2019-03-18. Final version published as submitted by the authors.